# Query da tradurre – Domenico Benfenati – 1 ora 5 richieste

## Query 1 – aggregazione mysql copilot avanzato

|  |  |
| --- | --- |
| ***Nome della Richiesta*** | Traduzione query aggregazione MySQL in Copilot |
| ***Descrizione del Problema*** | Devo tradurre una query di aggregazione da linguaggio SQL a linguaggio Python |
| ***Query da tradurre*** | **SELECT**  ae**.**TipoProdotto**,**  YEAR**(**ae**.**DataProduzione**)** **AS** AnnoProduzione**,**  **SUM(LENGTH(**ae**.**ManualeUtente**))** **AS** DimensioneTotaleManuali  **FROM** ApparecchiatureElettroniche ae  **GROUP** **BY** ae**.**TipoProdotto**,** YEAR**(**ae**.**DataProduzione**);** |
| ***Dettagli Aggiuntivi*** | Il database da utilizzare è MySQL, voglio suddividere il codice python in funzioni e gestire le eventuali eccezioni. Voglio includere dei commenti per ogni sezione di codice ottenuta. |
| ***Requisiti Tecnici*** | Usare la libreria mysql-connector, e dividere il codice in moduli separati. |
| ***Livello di Sicurezza*** | Non includere i dati sensibili all’interno del codice, ed evitare attacchi di tipo SQL-injection . |
| ***Extra*** | Voglio che l’output del codice venga mostrato come file json. Organizza i vari moduli in directory aggiungendo anche quella per i test e genera alcuni test per la verifica della sicurezza. |
| ***Struttura del codice*** | La struttura del codice è la seguente:  # import delle librerie utili  # definizione variabili globali  # definizione della funzione di connessione al database  # definizione delle funzioni utili per l’esecuzione  # main |

### Tabella metriche da compilare

|  |  |
| --- | --- |
| ***Tempo Generazione Risposta*** | 1:05 minuti |
| ***Numero Modifiche Senza Ausilio dello Strumento*** | 1 |
| ***Numero Modifiche Con Ausilio dello Strumento*** | 0 |
| ***Numero Errori*** | 1 |
| ***Tempo medio esecuzione script*** | 9.14 ms ± 644 µs |

Nota: la richiesta è soddisfatta, solo che per problemi di dati all’interno del database non è stato possibile testare la funzione per salvare l’output in formato json.

## Query 2 – aggregazione postgresql chatgpt avanzato

|  |  |
| --- | --- |
| ***Nome della Richiesta*** | Traduzione query aggrerazione PostGres in ChatGPT |
| ***Descrizione del Problema*** | Devo tradurre una query di aggregazione da linguaggio SQL a linguaggio Python |
| ***Query da tradurre*** | **SELECT**  ae**.**tipoprodotto**,**  **EXTRACT(**YEAR **FROM** ae**.**dataproduzione**)** **AS** annoproduzione**,**  **SUM(**OCTET\_LENGTH**(**ae**.**manualeutente**))** **AS** dimensionetotalemanuali  **FROM** apparecchiatureelettroniche ae  **GROUP** **BY** ae**.**tipoprodotto**,** **EXTRACT(**YEAR **FROM** ae**.**dataproduzione**);** |
| ***Dettagli Aggiuntivi*** | Il database da utilizzare è PostGres, voglio che generi il codice python in funzioni e che gestisci le eventuali eccezioni. Voglio dei commenti per ogni riga di codice che spiegano cosa viene fatto. |
| ***Requisiti Tecnici*** | Voglio suddividere la struttura del codice in moduli separati, e includere un connettore per il database PostGres. |
| ***Livello di Sicurezza*** | Voglio che gestisci le eventuali injection che possono capitare sul database PostGres. |
| ***Extra*** | Voglio che l’output della query venga proposto in un file csv, e che ogni modulo sia incluso in una cartella separata. Voglio che generi anche dei test per la query. |
| ***Struttura del codice*** |  |

### Tabella metriche da compilare

|  |  |
| --- | --- |
| ***Tempo Generazione Risposta*** | 28.84 secondi |
| ***Numero Modifiche Senza Ausilio dello Strumento*** | 0 |
| ***Numero Modifiche Con Ausilio dello Strumento*** | 0 |
| ***Numero Errori*** | 0 |
| ***Tempo medio esecuzione script*** |  |

## Query 3 – vista postgresql copilot intermedio

|  |  |
| --- | --- |
| ***Nome della Richiesta*** | Traduzione vista PostGres in Copilot |
| ***Descrizione del Problema*** | Devo tradurre una query per generare una vista da linguaggio SQL a linguaggio Python |
| ***Query da tradurre*** | **CREATE** **VIEW** vistamanipolazionedati **AS**  **SELECT**  ae**.**id **AS** apparecchiaturaid**,**  **CONCAT(**'Modello: '**,** ae**.**nomemodello**,** ' (ID: '**,** ae**.**id**,** ')'**)** **AS** descrizionemodello**,**  ae**.**numeroserie **AS** numeroserie**,**  **TO\_CHAR(**ae**.**dataproduzione**,** 'DD/MM/YYYY'**)** **AS** dataproduzioneformattata**,**  **UPPER(**ae**.**reparto**)** **AS** repartomaiuscolo**,**  **CONCAT(**'Fornitore: '**,** ae**.**fornitore**,** ', Acquisto: '**,** **TO\_CHAR(**ae**.**dataacquisto**,** 'DD/MM/YYYY'**))** **AS** dettagliacquisto**,**  **COALESCE(**ae**.**cliente**,** 'Nessun cliente'**)** **AS** clienteodefault**,**  **EXTRACT(**DAY **FROM** AGE**(**ae**.**garanziascadenza**,** **CURRENT\_DATE))** **AS** giornirimanentigaranzia**,**  **UPPER(**ae**.**statoproduzione**)** **AS** statoproduzionemaiuscolo**,**  **CONCAT(**'Dimensioni: '**,** ae**.**dimensioni**,** ', Peso: '**,** ae**.**peso**,** ' kg'**)** **AS** descrizionedimensionipeso  **FROM** apparecchiatureelettroniche ae**;** |
| ***Dettagli Aggiuntivi*** | Il database utilizzato è un database PostGres, voglio suddividere il codice in funzioni e commentarlo opportunamente. |
| ***Requisiti Tecnici*** | Voglio utilizzare un connettore per PostGres |
| ***Livello di Sicurezza*** | Stai attento alla sicurezza dei dati nella tabella |
| ***Extra*** |  |
| ***Struttura del codice*** |  |

### Tabella metriche da compilare

|  |  |
| --- | --- |
| ***Tempo Generazione Risposta*** | 2:20 minuti |
| ***Numero Modifiche Senza Ausilio dello Strumento*** | 0 |
| ***Numero Modifiche Con Ausilio dello Strumento*** | 0 |
| ***Numero Errori*** | 0 |
| ***Tempo medio esecuzione script*** | n/a |

Nota: non avendo inserito lo schema del codice, lo strumento è andato un po' in confusione ed ha generato un codice molto più lungo di quando doveva essere e anche ripetitivo in alcune funzioni

## Query 4 – join oracle google bard intermedio

|  |  |
| --- | --- |
| ***Nome della Richiesta*** | Traduzione query di join in Bard |
| ***Descrizione del Problema*** | Devo tradurre una query di join da linguaggio SQL a linguaggio Python |
| ***Query da tradurre*** | **SELECT**  PO**.**ID **AS** PRODOTTOORDINATOID**,**  PO**.**NUMEROORDINE**,**  PO**.**NOMEMODELLO**,**  PO**.**QUANTITA**,**  PO**.**PREZZOUNITARIO **AS** PREZZOUNITARIOPRODOTTO**,**  PO**.**VALUTA **AS** VALUTAPRODOTTO**,**  PO**.**DATAAGGIUNTA**,**  PO**.**DATACONSEGNAPREVISTA**,**  PO**.**STATOPRODOTTO**,**  PO**.**NUMERORMA**,**  PO**.**NOTE **AS** NOTEPRODOTTO**,**  AE**.**NOMEMODELLO **AS** NOMEMODELLOAPPARECCHIATURA**,**  AE**.**NUMEROSERIE**,**  AE**.**DATAPRODUZIONE**,**  AE**.**REPARTO**,**  AE**.**FORNITORE **AS** FORNITOREAPPARECCHIATURA**,**  AE**.**DATAACQUISTO**,**  AE**.**CLIENTE**,**  AE**.**REPARTOASSISTENZA**,**  AE**.**DATAULTIMAMANUTENZIONE**,**  AE**.**GARANZIASCADENZA**,**  AE**.**DESCRIZIONEPROBLEMA**,**  AE**.**COMPONENTEDIFETTOSO**,**  AE**.**TECNICOASSISTENZA**,**  AE**.**NOTE **AS** NOTEAPPARECCHIATURA**,**  OC**.**NUMEROORDINE **AS** NUMEROORDINECLIENTE**,**  OC**.**DATAORDINE**,**  OC**.**CLIENTE **AS** CLIENTEORDINE**,**  OC**.**INDIRIZZOSPEDIZIONE**,**  OC**.**CITTASPEDIZIONE**,**  OC**.**CAPSPEDIZIONE**,**  OC**.**PROVINCIASPEDIZIONE**,**  OC**.**NAZIONESPEDIZIONE**,**  OC**.**METODOPAGAMENTO**,**  OC**.**STATOPAGAMENTO**,**  OC**.**DATAPAGAMENTO**,**  OC**.**METODOSPEDIZIONE**,**  OC**.**DATASPEDIZIONE**,**  OC**.**STATOSPEDIZIONE**,**  OC**.**DATACONSEGNA**,**  OC**.**TOTALEORDINE**,**  OC**.**VALUTA **AS** VALUTAORDINE**,**  OC**.**NUMEROFATTURA**,**  OC**.**CODICEPROMOZIONALE**,**  OC**.**NOTE **AS** NOTEORDINE  **FROM** PRODOTTIORDINATI PO  **JOIN** APPARECCHIATUREELETTRONICHE AE **ON** PO**.**NOMEMODELLO **=** AE**.**NOMEMODELLO  **JOIN** ORDINICLIENTI OC **ON** PO**.**NUMEROORDINE **=** OC**.**NUMEROORDINE  **WHERE** **TO\_DATE(**  **TO\_CHAR(**PO**.**DATAAGGIUNTA**,** 'DD-MON-YYYY'**,** 'NLS\_DATE\_LANGUAGE=ITALIAN'**),**  'DD-MON-YYYY'  **)** **BETWEEN** **TO\_DATE(**'13-LUG-2017'**,** 'DD-MON-YYYY'**)** **AND** **TO\_DATE(**'30-GIU-2023'**,** 'DD-MON-YYYY'**)**  **AND** AE**.**DATAPRODUZIONE **<=** **TO\_DATE(**'30-GIU-2023'**,** 'DD-MON-YYYY'**)**  **AND** OC**.**DATAORDINE **>=** **TO\_DATE(**'01-GEN-2022'**,** 'DD-MON-YYYY'**);** |
| ***Dettagli Aggiuntivi*** | Voglio suddividere il codice in diversi moduli, e voglio che sia commentato |
| ***Requisiti Tecnici*** | Sfrutta un pattern architetturale MVC per la traduzione |
| ***Livello di Sicurezza*** | Voglio che l’accesso al database sia sicuro |
| ***Extra*** | Mostrami l’output in forma tabellare, e crea dei casi d’uso di test per la query |
| ***Struttura del codice*** |  |

### Tabella metriche da compilare

|  |  |
| --- | --- |
| ***Tempo Generazione Risposta*** | 8 sec |
| ***Numero Modifiche Senza Ausilio dello Strumento*** |  |
| ***Numero Modifiche Con Ausilio dello Strumento*** | 4 |
| ***Numero Errori*** |  |
| ***Tempo medio esecuzione script*** | n/a |

Nota: non ha generato il codice in prima battuta, sono servite 3 iterazioni per completarlo. Alla quarta iterazione google bard non è stato più in grado di rispondere e quindi abbiamo ritenuto il test fallito.

## Query 5 – select oracle copilot intermedio

|  |  |
| --- | --- |
| ***Nome della Richiesta*** | Traduzione query select Oracle in Copilot |
| ***Descrizione del Problema*** | Devo tradurre una query di select da linguaggio SQL a linguaggio Python |
| ***Query da tradurre*** | **SELECT** **\***  **FROM** "APPARECCHIATUREELETTRONICHE"  **WHERE** **TO\_NUMBER(SUBSTR(**"RISOLUZIONESCHERMO"**,** **INSTR(**"RISOLUZIONESCHERMO"**,** 'x'**)**+1**))** **>** 1080  **ORDER** **BY** "DATAPRODUZIONE" **DESC;** |
| ***Dettagli Aggiuntivi*** | Il database da utilizzare è Oracle, voglio che il codice che generi sia commentato e diviso in moduli separati |
| ***Requisiti Tecnici*** | Sfrutta un pattern architetturale di tipo MVC per la traduzione, e utilizza un connettore oracle per python per la connessione al database |
| ***Livello di Sicurezza*** | Voglio nascondere i dati sensibili all’utente, e che il codice sia resiliente agli attacchi esterni del tipo SQL-injection |
| ***Extra*** | Mostrami l’output in forma tabellare e crea degli esempi di test |
| ***Struttura del codice*** | Struttura del codice da generare:  # import delle librerie  # funzione di connessione per il database  # funzioni di utility per l’esecuzione del codice  # main |

### Tabella metriche da compilare

|  |  |
| --- | --- |
| ***Tempo Generazione Risposta*** | 59 secondi |
| ***Numero Modifiche Senza Ausilio dello Strumento*** | 1 |
| ***Numero Modifiche Con Ausilio dello Strumento*** | 1 |
| ***Numero Errori*** | 2 |
| ***Tempo medio esecuzione script*** | 155 ms ± 18.6 ms |

Nota: non ha generato in maniera appropriata i test, ha portato degli esempi per far vedere il funzionamento.